

VIDEO METRICS AND PARAMETERS FOR ACHIEVING OPTIMAL QUALITY IN LOW DELAY COMMUNICATION NETWORKS

YAKUBU SULEIMAN BAGUDA AND HANI ALJAHDALI

Department of Information Systems, King Abdulaziz University, Rabigh, Saudi Arabia.

Email: {ysuleman1, hmaljahdali}@kau.edu.sa

Revised May 2017, Revised July 2017

ABSTRACT. The need for optimal video quality over low delay communication networks is very crucial as their demand increases dramatically. Analysing, evaluating and exploring the codec metrics which can enhance the video quality when transporting the video over low delay communication networks has been a key issue of concern which requires thorough simulation and experimentation. Metrics such as quantization size, frame rate and frame size can significantly enhance the quality video. In this paper, the impacts of frame size, frame rate and quantization parameter on the compressed video quality has been investigated. More extensive simulations have been conducted in order to experiment different scenarios using various video samples at different frame sizes, quantization parameters, loss rate and encoding rate as well. More importantly, both low, medium and high complexity video samples have been examined to determine the impact of the aforementioned parameters on fundamental basic parameters which are essential toward yielding high quality video while compressing videos. The experimental results shows that increase in quantization parameter reduces the bits size and consequently leads to high compression.

Keywords: Advanced Video Coding (AVC), frame size, video encoder; video compression, video frame rate, quantization size, video quality

1. Introduction. In recent years, the world has witnessed an explosive growth in the video technology due to the advancement in video coding techniques. This has remarkably reshape the video industry and technology into a new dimension in which different applications and services are supported as a result of the new video compression techniques. The ability to compress video more and at the same maintaining video quality is extremely important. Initially, the MPEG-4 was designed to support low bit rate and interactive multimedia application. It has been equipped with functionalities such as scalability, error robustness and improved coding efficiency. Finally, MPEG 7 and MPEG 21 were targeted toward multimedia content description interface and entirely different from the conventional multimedia coding techniques. Each of the aforementioned coding standard provided standard syntax for encoding and decoding video stream. Therefore, high performance can be achieved by choosing the parameters which ultimately enhance the application performance. Each codec has special features and parameters. High compression efficiency and resilient to errors are important for video coding efficiency. H.264/AVC can effectively satisfy the aforementioned requirements [1][2]. Achieving low complexity, low delay and high video quality is challenging and difficult process especially in time varying channel condition. However, streaming application relies mainly on the encoded video stored in the streaming server. Hence, parameter setting used in encoding the video has vital role to play in increasing the error resiliency to dynamically changing environment.

Basically, the video encoder compresses raw video data by removing the temporal and spatial redundancies. The two major video standards used for compression are H.26X and MPEG. The international telecommunication union (ITU) develops the H.26x family and MPEG were developed by the international standard organization (ISO). The most recent development of highly sophisticated codec known as H.264/Advance Video Coding (AVC) was a joint work between ITU and ISO. Each standard normally specify the codec bit rate, complexity, resiliency and picture quality depending on the application which it is has been

designed for. H.26x family standards were developed primarily for supporting telecommunication applications, mobile wireless network and internet network delivery as well. H.261 was designed mainly to support video conferencing. It was followed by H.263 which was designed for low bit rate application but surprisingly has high performance at any bit rate when compared to H.261. The remarkable performance has led to the development of H.263+ and H.263++ which eventually have high compression efficiency and error resiliency. The latest codec on the H.26x family tree is the H.264, and it has better video quality when compared to all other video standard. Finally, the merger of H.264 with MPEG-4 leads to development of H.264/AVC which balanced the coding efficiency, implementation complexity and cost.

The remainder of the paper is organized as follows. Section 2 covers the video coding techniques, and Section 3 presents the simulation model. The result of the video codec simulation has been discussed in Section 4. Section 5 concludes the paper.

2. Overview of Video Codecs: The new video coding standard recommendation H.264 of ITU-T also known as International Standard 14496-10 or MPEG-4 part 10 Advanced Video Coding (AVC) of ISO/IEC is the latest standard in a sequence of the video coding standards H.261 (1990), MPEG-1 Video (1993), MPEG-2 Video (1994), H.263 (1995, 1997), MPEG-4 Visual or part 2 (1998) [3]. These previous standards reflect the technological progress in video compression and the adaptation of video coding to different applications and networks. The development H.264/AVC has had a profound impact on multimedia communication industry due to its performance and reliability in terms of supporting multimedia applications in various types of networks.

The H.264/AVC was developed by the Joint video team (JVT) [4][5]. It is a collaborating task between the ITU and ISO. High compression and others unique features have made it superb and an extremely important tool for video communication [6]. The H.264/AVC is aimed at applications which primarily used MPEG2, H.263 and MPEG-4 part 2. The capabilities of the aforementioned coding techniques have been adapted and enhanced. This makes H.264/AVC more sophisticated when compared to other video coding techniques.

Table 1. Comparisons between H.264 and other CODECs [7]

Type	Standard Organization	Affordable Data rate	Application
JPEG	ISO/ITU-T		Image
MPEG 1	ISO	1-2 kbps	Video CD
MPEG 2	ISO	4-30 kbps	DTV, SDTV, HDTV
MPEG 4	ISO	24-1024 kbps	Interactive video
H.261	ITU-T	80-320 kbps	Video conferencing
H.262, H.262++	ITU-T		DTV, SDTV
H.263, H.263++	ITU-T	10-2048 kbps	Videophone
H.264	JVT	Variable	Improve video compression

Table 1 summarizes the different coding techniques and their respective applications. It is very obvious that H.264 has many advantages when compared to the other existing codec's. It serves up to 50% bit rate when compared to MPEG-1 and MPEG-2 [3]. More importantly, it produces high quality video at low rate. Its error resilience feature and network adaptation suits the wireless environment where there is high tendency for packet loss and bit error rate. These features make H.264 ideal standard for wireless video streaming application where the channel is time varying and dynamic.

The input video is firstly divided into macroblocks and each macroblock has three components namely Y, Cr and Cb. This represents the luminance and chrominance components. The chrominance signals are subsamples by a factor of 2 in horizontal and vertical directions. The macroblock consists of 16X16 picture element representing the luminance and two blocks of 8x8 picture elements for the chrominance components. The macroblocks are further coded into intra and inter mode. The macroblock is predicted in inter mode using motion compensation. The displacement vector for each block is estimated and transmitted. Also, the difference between the original and predicted block is transformed, quantized and coded using entropy coding. The H.264 uses integer transform which eventually approximate roughly the discrete cosine transform used in

MPEG-2 without mismatching problem. Hence, the difference results to prediction error. The encoded video is decoded by taking the inverse transform of the quantization coefficient and adding it to the prediction signal. It is very important to note that the macroblocks are stored in memory in raster scan order.

Each video frame is presented in form of macroblock. Motion compensation and prediction plays an important role in encoder. Motion estimation is used to determine the region from the previous frame that best matches the macroblock in the current frame. Alternatively, it determines the pixel blocks which contain information representing parts of the image that have moved and the distance they have moved. Motion compensation is a process of reconstructing image using the segment of the previous image and information about motion and it require low computational time [8]. The next step is transformation in order to ensure that the transform domain is compacted. Basically, discrete cosine transform (DCT) is used due to its less computational complexity and high coding capability. By removing the temporal and spatial information results in loss of information and eventually the video is compressed. Huffman or arithmetic coding is generally used to quantize the coefficients.

It is obvious that the compression efficiency increases with the increase in cost of complexity. The discrete cosine transforms normally convert the block of pixels into block of transform coefficient. Each coefficient represents spatial frequency components of the original block. For instance, two dimensional DCT with $n \times n$ block of pixels can be described mathematically by the expression in equation (1) and (2) below.

$$F(u, v) = \frac{2}{n} C(u) \cdot C(v) \cdot \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} f(i, j) \cos\left(\frac{(2i+1)uf}{2n}\right) \cos\left(\frac{(2j+1)vf}{2n}\right) \quad (1)$$

$$\text{Where } C(x) = \begin{cases} \frac{1}{\sqrt{2}}, & x = 0 \\ 1, & \text{otherwise} \end{cases} \quad (2)$$

In a nutshell, the quantization process involves compressing the spatial frequency representation of the blocks. Normally the quantized DCT coefficients can be expressed mathematically by equation (3).

$$I(u, v) = \begin{cases} 0 & \text{for } |F(u, v)| < T \\ \left\lceil \frac{F(u, v)}{2Q} \right\rceil & \text{for } |F(u, v)| \geq T \end{cases} \quad (3)$$

However, the quantization scale can be used to adjust the encoding which is extremely useful in time-varying environment such as wireless network. This can be applicable in adjusting the video frame size to suit the available bandwidth for real time video application such as video conferencing.

The formats of all video sequence used for this work are in YUV format. This is primarily due to the fact that human visual system (HVS) is more sensitive to luminance (intensity) than colour. Most commonly used colour sub-samples have been shown in figure 1. As it can be seen that the luminance are represented by circles while the chrominance represented by Y. It is very important to note that 4:4:4 and 4:2:2 sub-samples are commonly used in high quality colour reproduction. 4:2:0 sub-sample is widely applicable in video conferencing and digital television. In this work, 4:2:0 have been used for all the experimentations primarily due to the fact it is extensively used for modern video coding standards [9].

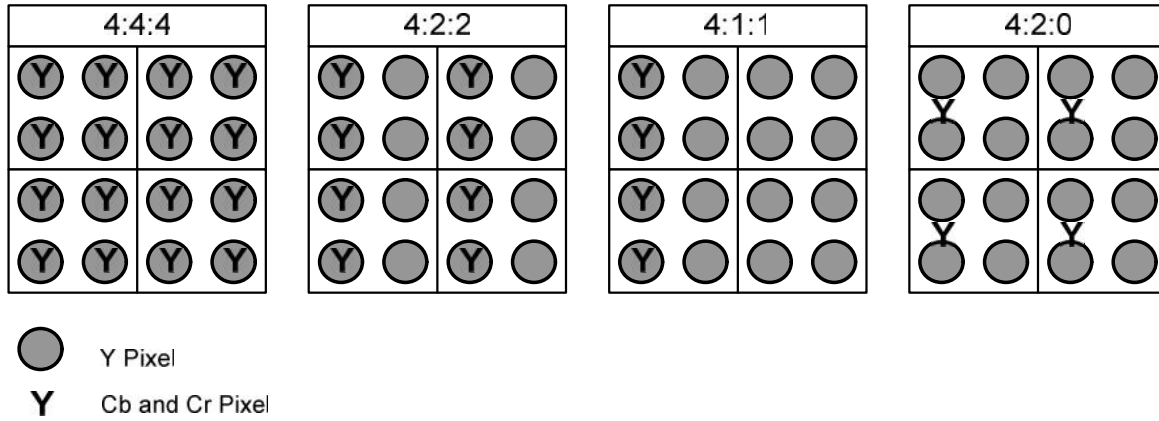


Figure 1 Colour sub-samples

As can be seen from Table 2, CIF and QCIF are the most frequently used video format for wireless transmission. In this work, different raw video sequence in QCIF formats have been used and both eventually have the same width-height ratio. The development of H.264 has led to more compression efficiency as well as the capability to adapt to the network transmission requirement [10]. This network adaptation capability will be useful in wireless network in order to maintain the video quality. More importantly, the H.264 has the ability to refresh pictures in macroblock level if there is any loss or change in bit stream.

Table 2. CIF & QCIF Video Formats

Video Format	Columns (Y)	Rows (Y)	Columns (U,V)	Rows (U,V)	Frame rate (Hz)
CIF	352	288	176	144	30
QCIF	176	144	88	72	30

3. Experimental Simulation: The video encoding and testing model is used offline to encode and decode the video samples. This is primarily due to the complexity of encoding and decoding while simulating the scenarios. Different experiments have been conducted using the model shown below in figure 3. It describes the procedure used to test the six (6) video sequences based on different encoding parameters and setting. Basically, the video model consists of the reference software and video samples used for the experimentation. The video samples used include *Coastguard*, *Foreman*, *Akiyo*, *News*, *Hall* and *Mobile* as shown in Figure 2. Each video sample has a total of 300 frames and in QCIF format.

The H.264 reference software for research purposes can be freely obtained from [11]. The parameters are set based on the requirement of the application. After the parameters have been set, the reference software is run and the targeted video sample is encoded. The encoded video stream is stored in dot 264 (.264) format within the H.264 reference software package. Different files are created which contain information about the encoded video statistics. The captured information about the encoded video file is stored in stats and log files. It is mainly based on the information contain in these files that the encoded video is decoded. The video encoding and testing simulation model is as shown in Figure 3 below.

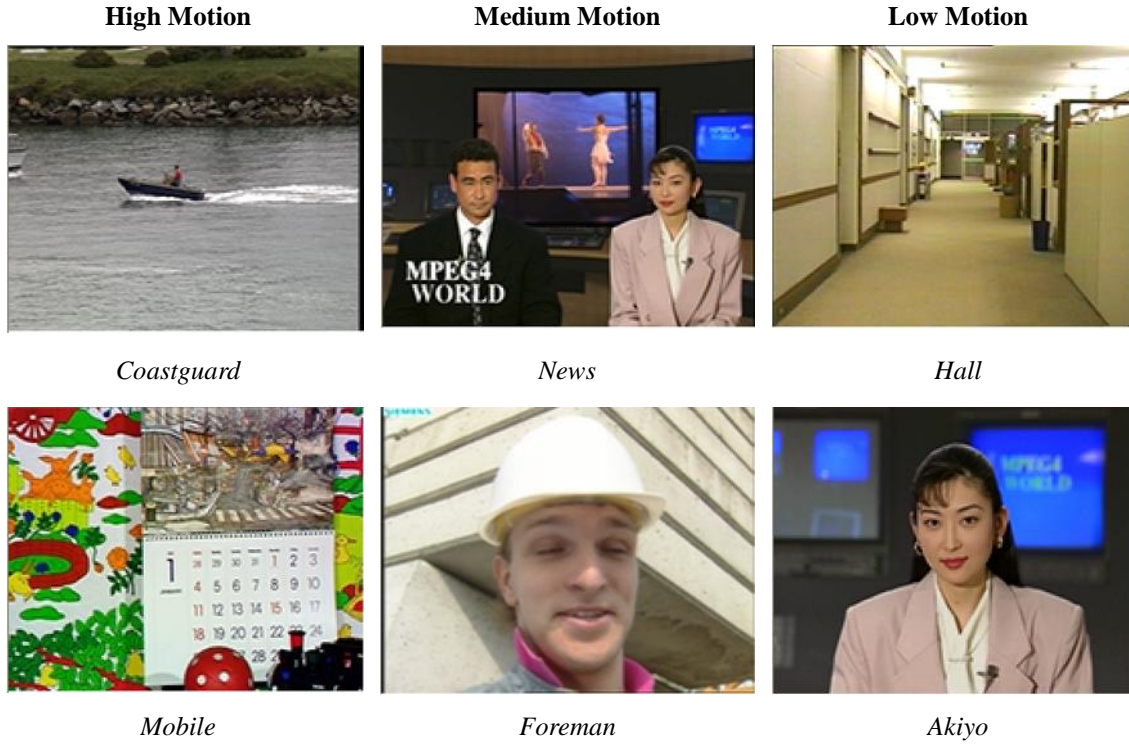


Figure 2 QCIF (176X144) Video test sequences

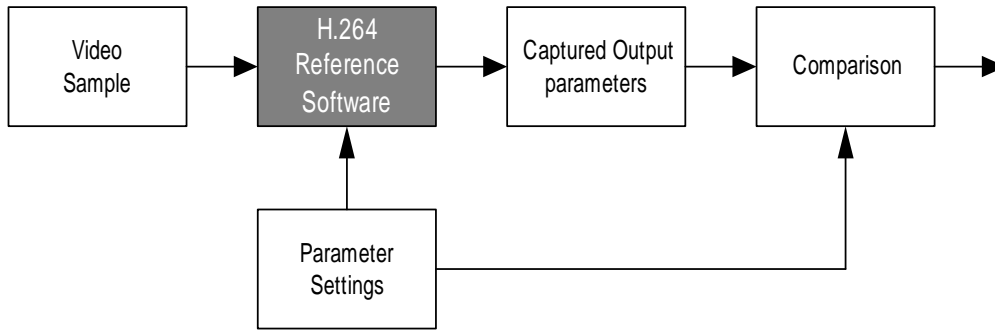


Figure 3 Video encoding and testing model

H.264/AVC is used in encoding and decoding the video samples for the wireless video streaming. The nature how the video samples are encoded plays an important role in maintaining the video streaming quality especially with the rapid variation in channel condition. The video model used for the experiment is shown in figure 4 which each group consists of an I-frame and 14 P-frame. All of the six QCIF video samples are encoded with the same group structure for the video streaming application. Therefore, the group structure used is IPPPPPPPPPPPPPP. I and P-frames were encoded using the quantization factor of 25 and 22 respectively ($I=25$, $P=22$). The size of I-frame is very much higher when compared to P-frame primarily due to many parameters included in the frame, i.e motion vector, motion compensation, and motion estimation [3]. The first frame in the group of frame is represented as i . The group can be represented by $G = \{i, i + 1, \dots, i + n\}$.

It is very important to note that the first captured frame is encoded in INTRA mode (I-frame) and the blocks in the subsequent frames are encoded in INTER mode (P-frame). The reason is mainly for coding efficiency where the structure used in this experiment is suitable for real-time encoding. Therefore, the use of B-frame for real time encoding is not very important. This is due to the fact that the encoding of consecutive frames is not sequential and leads to additional delay.

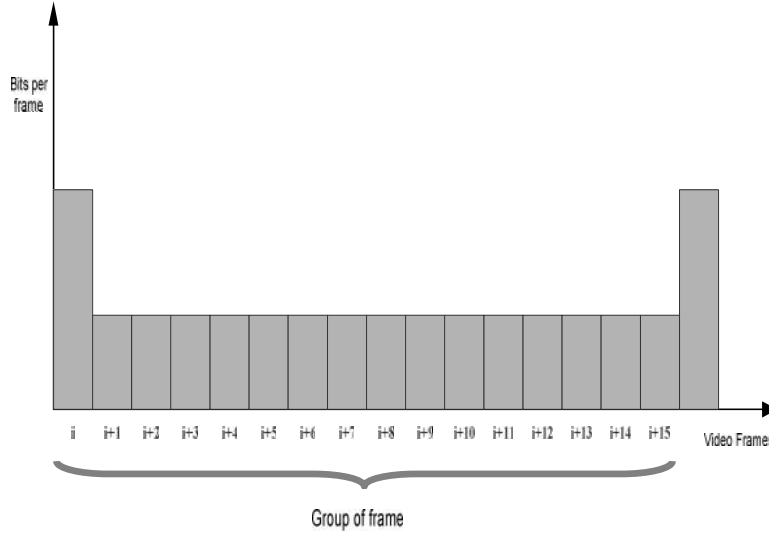


Figure 4 Video bit stream and their size

Figure 4 shows the video format of the frames used in the simulation. Each frames is displayed using the rate mentioned previously which eventually creates the motion. The number of frames display in second in continuous motion will appear to the human eye as moving object due to persistent of human vision. For this research, the NTSC format is used throughout the experiment unless otherwise stated. The frame rate for the standard is approximately 30 fps. Each individual frame consists of pixels which depend on the type of frame format. The ITU-R/CCIR-601 Quarter Common Intermediate Format (QCIF) used throughout this research work has 176x144 pixels.

4. Results & Discussion. The experimental study is aimed at identifying the influence of quantization parameter to encode the video sequences. In this study, the six (6) video samples were encoded at a rate 310 Kbps. The quantization parameter was varied from 22 to 42 and the size for both I and P frame were observed. As shown in figure 5, the frame size decreases with increase in quantization. This has profound impact on the quality as well. Higher quantization parameter can eventually lead to low video quality and high compression.

It is very obvious that the size of I-frame is bigger than P-frame primarily due to the fact that it contains more information about the group of picture (GOP). Parameters such as motion vector and compensation are included in the I-frame and hence its size is expectedly bigger compared to other frame. The graph below shows the relationship between the frame size and quantization parameter for the video sequences. It can be seen that the number of bits representing the I and P frames for high motion (COASTGUARD and MOBILE) video sequence is higher when compared to medium and low motion video sequence. Hence, the high motion video sequence can be identified from the graph based on the I-frame size.

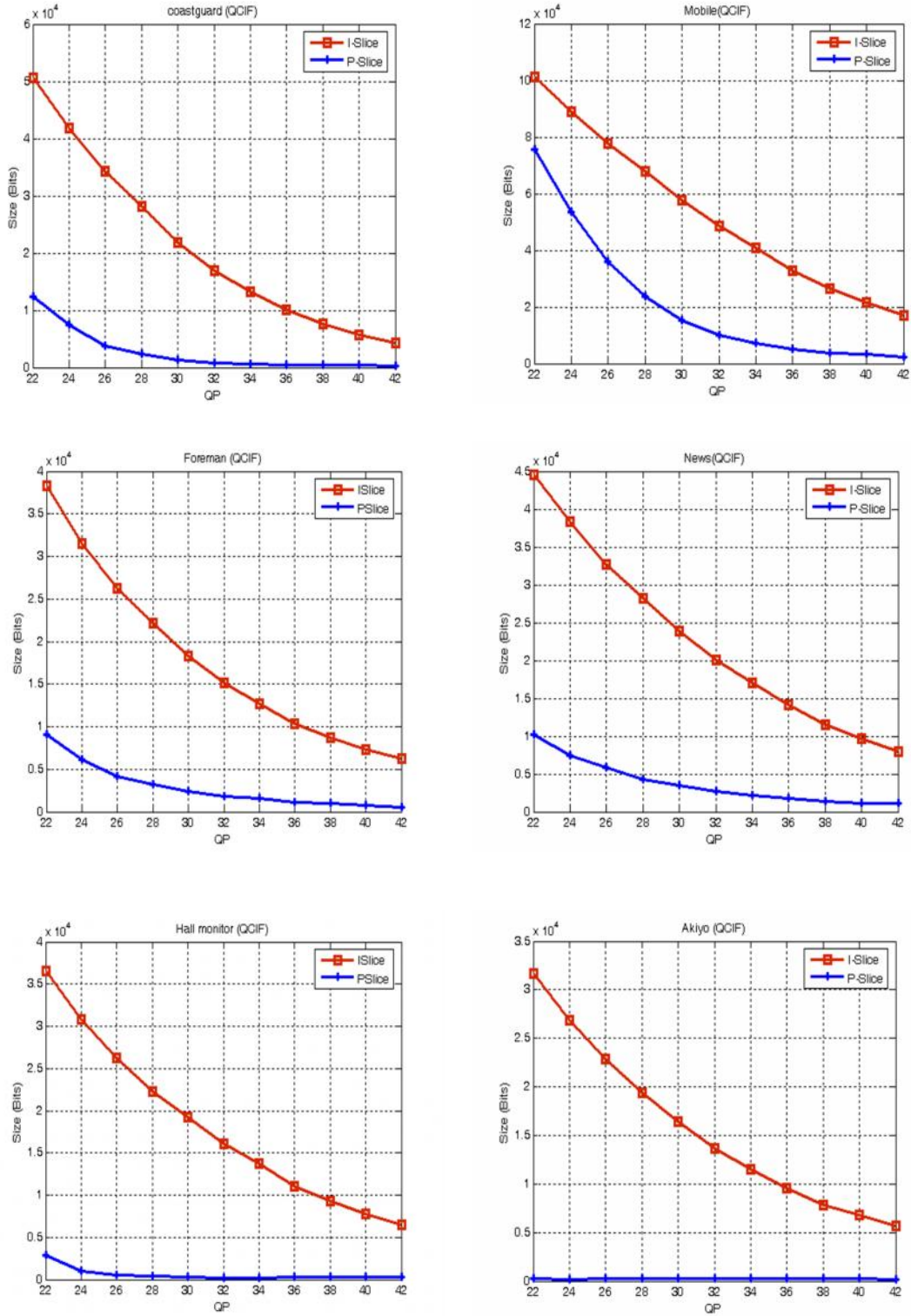
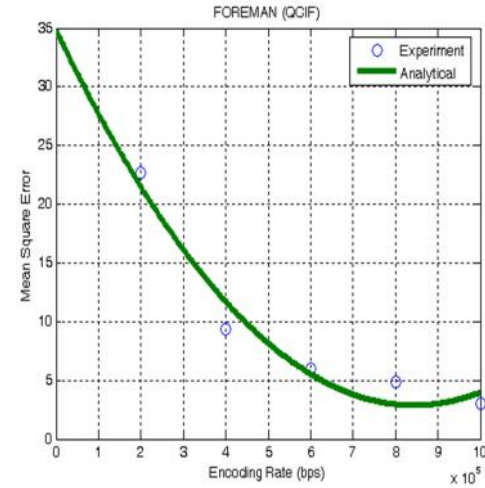
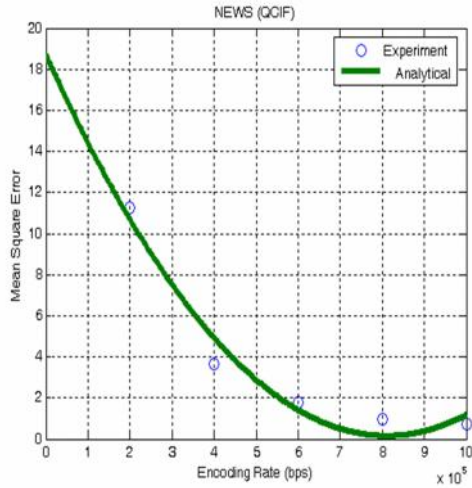
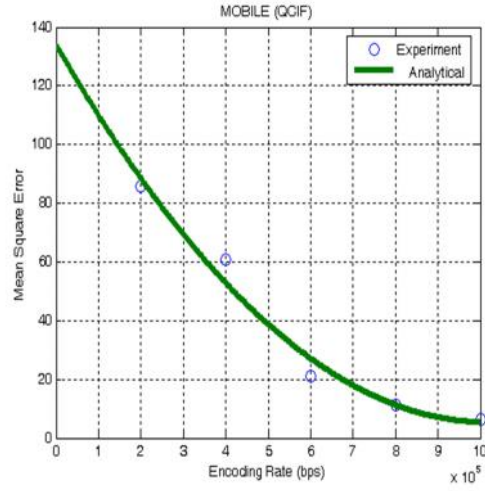
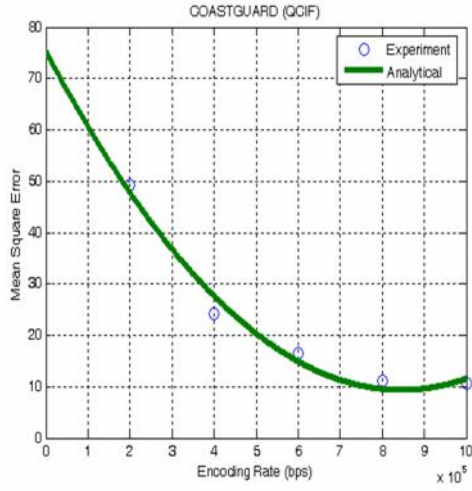


Figure 5 I and P-slice size for the sample sequence at different quantization parameter

A. Encoding Rate

The impact of the encoding rate on the mean square error has been studied in this section. The mean square error (MSE) is indirectly proportional to the peak signal to noise ratio (PSNR) as shown in $10 \log_{10} \left(\frac{255^2}{MSE} \right)$. Basically, PSNR can be used to determine the quality of a particular video sequence. The H.264 encoder was used to encode the video sequence at different rates and the resultant MSE was observed. It can be seen from figure 6 that the MSE is significantly reduced as the encoding rate increases. The encoding rate was varied from 100kbps to 1Mbps.

It is very important to consider the tradeoff between computational power and high encoding rate due to power limit in the devices. A better video quality can be achieved by encoding the video stream at high rate at the expense of computational power. Also, it is very important to note that high encoding rate in highly erroneous channel can lead to poor video quality. As illustrated in figure 5, both experimental and analytical results have almost the same trend, but best fitness has been used to estimate of the trend analytically based on the experimental results. It is very important to note that encoding power consumption is always proportional to the encoder complexity. Since 384 Kbps is the best attainable rate for video streaming application, the encoding rate should be lesser in order to prevent any over flow in buffer especially for live wireless video streaming. All the six (6) video sequences have been encoded at the rate of 310 kbps.



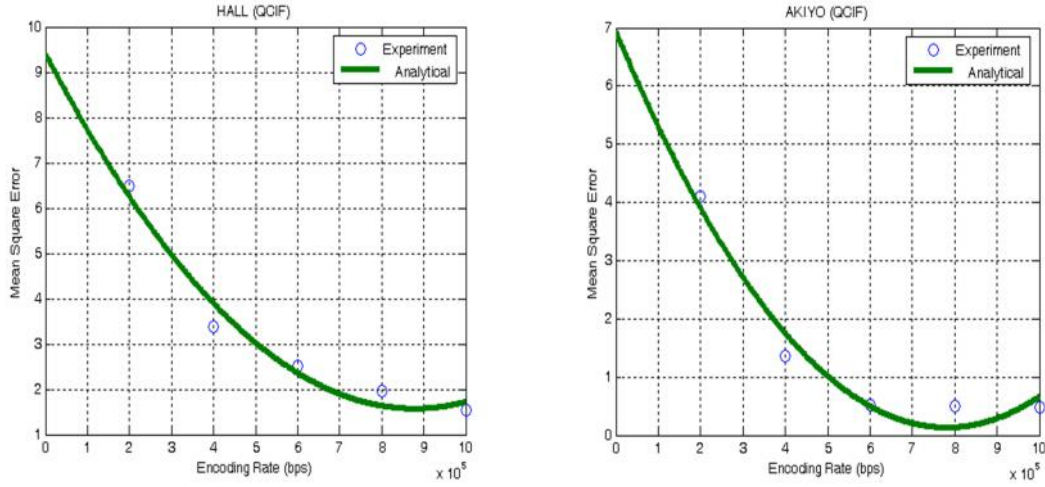


Figure 6 Mean Square Error (MSE) for the sample sequence at different encoding rate

B. Video Encoding Loss Rate

In order to critically analyse and examine the impact of encoding loss rate on the MSE, the total loss in video quality as a result of encoding, transmission and decoding need to be considered. More importantly, the MSE is indirectly proportional to the video quality. Better video quality can be achieved with low loss rate while compressing the video data. The range is selected accordingly to determine its effects on the video quality as it significantly increases. The regression constant is then computed using the best fit line. This will help tremendously in estimating the MSE at any particular time once the loss rate is known. Therefore, selecting suitable loss rate while encoding the video stream has a greater impact on the video quality is clearly shown in figure 7. More importantly, the graphs illustrate how each of the samples varies with in the loss rate. Both loss rate and MSE are directly proportional to one another. Initially, the loss rate has been set to 0.02 (2%) and adjusted at an interval of 0.02.

Initially, the loss rate for each of the six video sequence used in the experiment is studied intensively in order to determine its impact on the video quality. A simple group-of-pictures (GOP) structure of IPPPPPPPPP was used for the experiment, and decoding a P-frame requires the decoding of previous frames in the same GOP is to achieve relatively low compression. As shown in figure 7, some sequences are more susceptible to loss rate than others depending on the nature of the video sequence. It is very important to note all the video sequences have been encoded at the same rate. In figure 7, it can be clearly seen that loss due to encoding has dramatic impact on video quality and its impact depend greatly on the type of video sample. The variation in MSE changes abruptly due to the randomness of the loss, but best fit curve has been used to determine the characteristic of each video sample. The lower the MSE and the better the video quality (PSNR). This has been used primarily in preparing the samples for the video streaming application. For live video streaming, it can be used in controlling the video encoding quality.

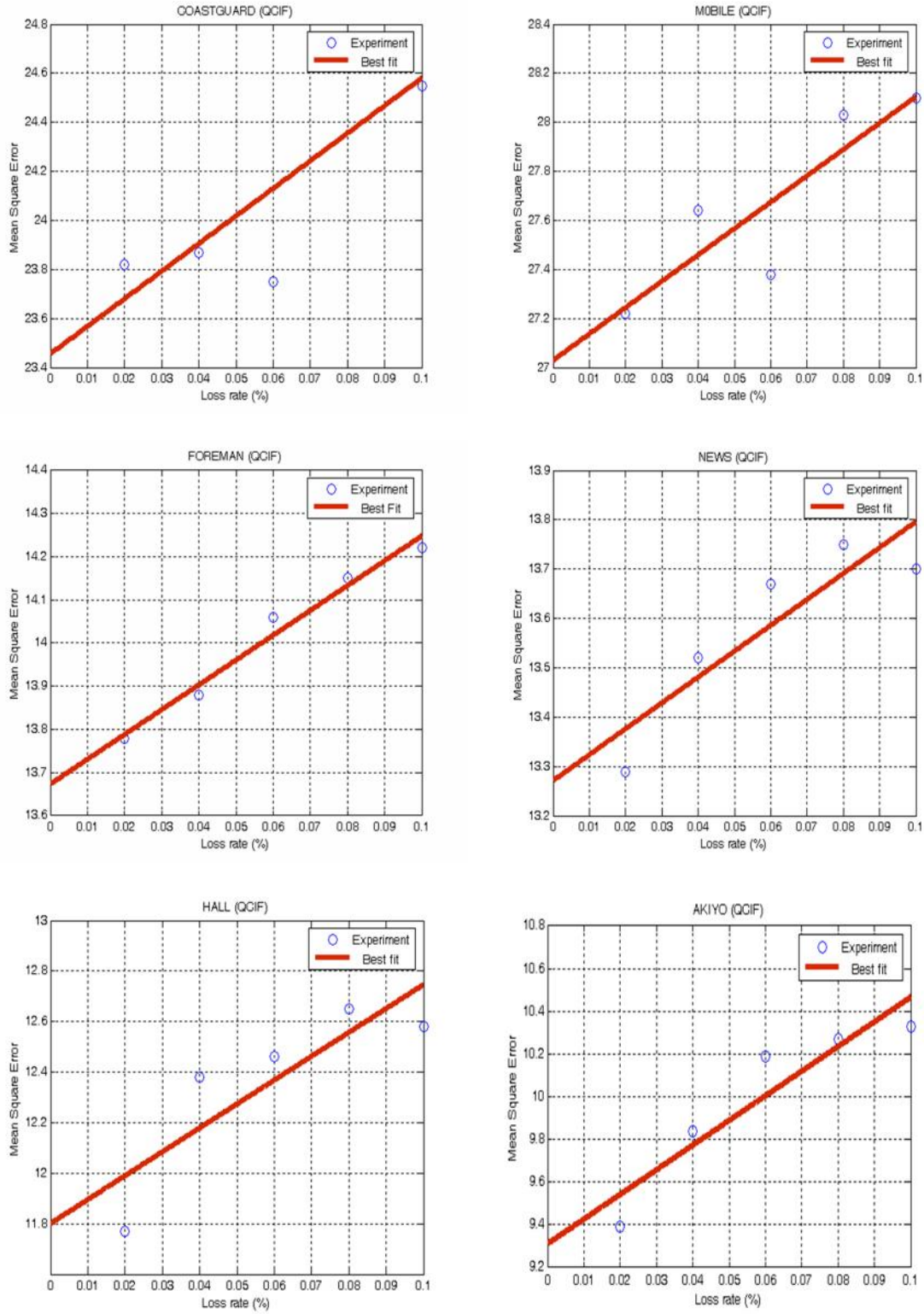


Figure 7 Impact of loss rate on means square error for different sequence

Conclusion: In this paper, the result for the experimentation and analysis for the video coding has been presented in order to show the effect of the quantization parameter on frame rate and size which are important in preserving the video quality. More rigorous analysis of various video test sample with different complexity has shown how the variation of quantization parameter, frame size and frame rate significantly effect the videos under examination. The study will eventually enable the development of models for prediction of rate, quality and power consumption which extremely useful for various video applications and services in low energy devices. Our future works will focus primarily on the development of nature-inspired energy efficient rate-power consumption models specifically design for low delay applications and devices.

Acknowledgement: The author would like to thank all those who contributed toward making this research successful. Also, I would like to thanks to all the reviewers for their insightful and valuable comment. This work was supported by the Deanship of Scientific Research (DSR), King Abdulaziz University, Saudi Arabia, under grant No. 830-64-D1437. The authors therefore gratefully acknowledge the DSR technical and financial support.

REFERENCES

- [1]. Kwon, S. K., Punchihewa, A., Bailey, D. G., Kim, S. W., & Lee, J. (2012). Adaptive simplification of prediction modes for H. 264 intra-picture coding. *IEEE Transactions on Broadcasting*, 58(1), 125-129.
- [2]. Huang, Y. H., Ou, T. S., & Chen, H. H. (2010). Fast decision of block size, prediction mode, and intra block for H. 264 intra prediction. *IEEE transactions on circuits and systems for video technology*, 20(8), 1122-1132.
- [3]. Ostermann, J., Bormans, J., List, P., Marpe, D., Narroschke, M., Pereira, F., ... & Wedi, T. (2004). Video coding with H. 264/AVC: tools, performance, and complexity. *IEEE Circuits and Systems magazine*, 4(1), 7-28.
- [4]. Marpe, D., Wiegand, T., & Sullivan, G. J. (2006). The H. 264/MPEG4 advanced video coding standard and its applications. *IEEE communications magazine*, 44(8), 134-143.
- [5]. Marpe, D., Wiegand, T., & Gordon, S. (2005). H. 264/MPEG4-AVC fidelity range extensions: tools, profiles, performance, and application areas. In *Image Processing, 2005. ICIP 2005. IEEE International Conference on* (Vol. 1, pp. I-593). IEEE.
- [6]. Schäfer, R., Schwarz, H., Marpe, D., Schierl, T., & Wiegand, T. (2006). MCTF and scalability extension of H. 264/AVC and its application to video transmission, storage, and surveillance. In *Visual Communications and Image Processing 2005* (Vol. 5960, p. 596011). International Society for Optics and Photonics.
- [7]. Sullivan, G. J., & Wiegand, T. (1998). Rate-distortion optimization for video compression. *IEEE signal processing magazine*, 15(6), 74-90.
- [8]. Mattison, P. E. (1994). *Practical digital video programming with examples in C*. John Wiley & Sons, Inc..
- [9]. Richardson, I. E. (2004). *H. 264 and MPEG-4 video compression: video coding for next-generation multimedia*. John Wiley & Sons.
- [10]. Sullivan, G. J., Wiegand, T., & Stockhammer, T. (2001). Using the draft H. 26L video coding standard for mobile applications. In *IEEE International Conference on Image Processing* (Vol. 3, pp. 573-576).
- [11]. H.264/AVC Reference Software, [Online]. Available: <http://iphome.hhi.de/suehring/>